REAL ACADEMIA
DE INGENIERÍA

PROCEEDINGS
OF THE

ADVANCE AND APPLICATIONS OF DATA SCIENCE AND ENGINEERING

Madrid, June 14th-17th

Editors: Harold Molina-Bulla
Irene Córdoba Sanchez

Organized by

Real Academia de Ingeniería

REAL ACADEMIA
DE INGENIERÍA

with the collaboration of
- Data Science and Engineering Consortium
- Project CASI-CAM-CM, Comunidad de Madrid

CASI-CAM-CM
CONCEPTOS Y APLICACIONES DE LOS SISTEMAS INTELIGENTES
(S2013/ICE2845)

Comunidad de Madrid

Universidad Carlos III de Madrid

POLITÉCNICA

UAM

UNIÓN EUROPEA
Fondos Estructurales

- DAMA Network, Ministerio de Economía y Competitividad

GOBIERNO DE ESPAÑA

MINISTERIO DE ECONOMÍA Y COMPETITIVIDAD

Real Academia de Ingeniería

Advances and Applications of Data Science & Engineering

An International workshop organized with the collaboration of
Data Science and Engineering Consortium
Project CASI-CAM-CM, Comunidad de Madrid
DAMA Network, Ministry of Economy and Competitiveness

Presentation

Reality speaks data, and Science and Engineering help us to understand its messages. This workshop consists of two series of activities. Mornings are dedicated to advances in data processing and their application to a number of relevant fields. Meetings of three research networks (Data Sci&Eng Consortium, CASI-CAM-CM, and DAMA) will be celebrated after the lunch time. The purpose is to promote both scientific knowledge and collaboration among different agents for getting benefit from data.

Editors:

- Harold Molina-Bulla
- Irene Cordoba Sánchez

FOREWORD

PROCEEDINGS

SCIENTIFIC PROGRAM

Tuesday, June 14th

09:30 h

Opening Ceremony

J.M. Torralba-Castelló, D. General UI, CE-CM

10:00 h

Session 1A: ***Deep Learnin***

Chair: A. R. Figueiras-Vidal (RAIng, UC3M/ DS&EC, CASI-CAM-CM, DAMA)

Invited lecture: *A (concise) tutorial on deep learning* A. R. Figueiras-Vidal (RAIng, UC3M/ DS&EC, CASI-CAM-CM, DAMA)

Contributions

1. *Data analysis of deep learning representations,* D. García-Gasulla (UPC/ DAMA)
2. *Partial boosting of deep stacking networks,* M. Montoya-Catalá, A.R. Figueiras-Vidal (UC3M/ DS&EC, CASI-CAM-CM, DAMA)
3. *Information based approaches for Bayesian optimization,* E. Garrido, D. Hernández-Lobato (UAM/ CASI-CAM-CM, DAMA)

11:45 h

Session 1B: ***Applications to Chemistry, Pharmacy, and Biology***

Chair: M. H. Hayes (GMU, USA/ DS&EC)

Invited lecture: *Bayesian networks in Neuroscience*

P. Larrañaga (UPM/ CASI-CAM-CM)

Contributions

4. *Automatic classification of neural morphologies,* B. Mihaljevic (UPM/ CASI-CAM-CM)

Wednesday, June 15th

09:30 h

Session 2A: ***Security and Safety***

Chair: B. Vitoriano (UCM/ CASI-CAM-CM)

Invited lecture: *Differential analysis as a data science tool for cyber security*

J.H. Jones, Jr. (GMU, USA/ DS&EC)

Contributions

5. *SEDD: A soft data-science approach for assessing consequences of natural disasters,* J. Tinguaro-Rodríguez, B. Vitoriano, J. Montero (UCM/CASI-CAM-CM)
6. *An Architecture for Risk Management Decisions in Aviation Safety at State Level,* D. Ríos (AXA-ICMAT Ch./aCASI-CAM-CM), P. H. Coronado, F. Bernal (Ag. Esp. Seguridad A.), J. Gómez, C. Alfaro (Spain R. Academia de Ciencias)
7. *Geometric models for video surveillance in road environments: Vehicle tailgating detection,* E. Pla-Sacristán, I. González-Díaz, F. Díaz-de María (UC3M)

11:15 h

Session 2B: *Cognitive Health*

Chair: F. García-Nocetti (UNAM/ DS&EC)

Invited lecture: *The evolution of healthcare: First steps towards cognitive healthcare*

A. Solanas (URV)

Contributions

8. *Active sensing in human activity recognition,* A. Nazábal, A. Artés-Rodríguez (UC3M/ CASI-CAM-CM)
9. *Machine learning methods in electronic nose analysis with Deep Architectures,* I. Rodríguez-Luján, J. Fonollosa, R. Huerta (UAM/ CASI-CAM-CM, DAMA)
10. *Decoding for neural protheses*, H. Dantas, V.J. Mathews (Oregon SU., USA)

12:30 h

DS&EC, CASI-CAM-CM, and DAMA open session

Thursday, June 16th

10:00 h

Invited Presentations

- *International cooperation in Horizon 2020,* E. Pelayo-Campillos (Natl. Contact Point ICT-H2020, CDTI)
- *Spanish international R&D cooperative programs with USA & Latin America,* E. Iglesias Cadarso (Natl. IBEROEKA Coord., CDTI)

11:15 h

Session 3A: *Smart Cities*

Chair: M. Mucientes (USC/ DS&EC, DAMA)

Invited lecture: *A guideline for a public-private partnership on urban big data sharing*

D. Sarasa-Funes (UPM/ Ay. Zaragoza/ aCASI-CAM-CM)

Contributions

11. *Enhancing smart mobility: some Artificial Intelligence solutions*, J. Vázquez-Salceda (UPC-DAMA)

12:30 h

Session 3B: *Other Applications*

Chair: J. Dorronsoro (UAM/ CASI-CAM-CM, DAMA)

Contributions

12. *Prediction of building temperatures for energy optimization,* P. Rodríguez-Mier, M. Fresquet, M. Mucientes, A. Bugarín (USC/ DS&EC, DAMA)
13. *Data science in sports and the application of machine learning for jump shot prediction in the NBA,* R. Alaghbar, M. Hayes (GMU, USA/ DS&EC)

Friday, June 17th

10:00 h

Session 4: *Singular Problems*

Chair: C. Hervás (UCO/ DAMA)

Invited lecture: *Real time data mining: Data mining for the XXI Century*

J. Gama (U. Porto, Portugal)

Contributions

14. *Dynamic learning of Cases for data streams,* F. Orduña-Cabrera, M. Sànchez-Marrè (UPC-DAMA)
15. *An exploratory evaluation of Bayesian principled approaches to solve imbalanced problems,* S. Roca-Sotelo, A. R. Figueiras-Vidal (UC3M/ DS&EC, CASI-CAM-CM, DAMA)
16. *Machine learning decomposition models for partial ordering problems: An application to melanoma severity classification,* J. Sánchez-Monedero, M. Pérez-Ortíz, A. Sáez, P.A. Gutiérrez, C. Hervás

Prediction of building temperatures for energy optimization

Pablo Rodríguez-Mier*, Marc Fresquet*, Manuel Mucientes*, Alberto Bugarín*
*Centro Singular de Investigación en Tecnoloxías da Información (CiTIUS), Universidade de Santiago de Compostela
Santiago de Compostela, Spain
Email: pablo.rodriguez.mier@usc.es, marc.fresquet@rai.usc.es, manuel.mucientes@usc.es, alberto.bugarin.diz@usc.es

***Abstract*—The reduction of energy consumption in buildings is one of the goals to improve energy efficiency. One way to achieve energy savings in buildings is to develop intelligent control strategies for heating systems that are able to reduce power consumption without affecting the thermal comfort. An intelligent control system must be able to predict the temperature of the building in order to manage the heating system. In this paper, we present three rule-based models that are able to predict the indoor temperature. The models have been learned with FRULER, a genetic fuzzy system that generates accurate and simple knowledge bases. Our approach has been validated with real data from a residential college showing errors lower than 0.50°C in the prediction of the temperatures.**

I. Introduction

Buildings account for 40% of the total energy consumption in the EU, according to European Directive 2010/31/EU on energy efficiency in buildings. Since the expansion this sector is currently experiencing will cause an inevitable rise of that percentage, it seems clear that the reduction of energy consumption and the use of energy from renewable sources in the building sector will play a key role in the measures taken to reduce emissions of greenhouse gases.

One way to achieve energy savings in buildings is by reducing the total working hours of heating systems. However, a decrease in the total usage may lead to important decreases of indoor temperatures that can affect thermal comfort. In order to prevent this, automatic heating control systems must predict the future indoor temperature for a particular control policy in order to find the best strategy that minimizes power consumptions while keeping thermal comfort.

Current methods for indoor temperature prediction [2] are mostly based on physical model simulations [12] and black-box machine learning methods [6], [13], [1], [11]. Physical models describe the building behaviour by solving theoretical equations that describe to a certain precision the different dynamics and interactions between the variables. Although these methods are very powerful to simulate the different dynamics of a building, especially when there is no real data available, in general these methods are: 1) very time-consuming since they require many simulation hours, which prevents their application for predicting temperatures in small temporal windows; and 2) complex to formulate, since it is very difficult to produce a detailed model of a complex building, especially when there are many unknown factors that can affect the temperature dynamics. On the other hand, machine learning models can overcome some of these limitations by learning the behaviour from real data. However, current techniques, which are mostly black-box models based on neural networks, are hard to interpret and thus the interaction of the different variables of the building remains unknown.

This research was supported by the European Union LIFE programme (grant LIFE12 ENV/ES/001173), the Spanish Ministry of Economy and Competitiveness (grant TIN2014-56633-C3-1-R) and the Galician Ministry of Education (grants CN2012/151 and GRC2014/030). All grants were co-funded by the European Regional Development Fund (FEDER program).

In this sense, the generation of accurate and interpretable models for indoor temperature prediction is fundamental for 1) modelling the energy-building behaviour and 2) discovering which are the most relevant variables that affect the indoor building temperature and are related to power consumption. Within this context, initiatives such as the EU LIFE-OPERE project [4], where this research is framed, have started. OPERE has among its goals the setting of efficient management systems in energy networks, both thermal and electrical, in existing installations with large energy consumption.

In this paper, a rule-based regression model for indoor temperature prediction is proposed. The aim is to build and validate prediction simple and accurate models. To do so, we have modelled the indoor temperatures of a number of of a residential college using the FRULER Genetic Fuzzy System (GFS) [9]. The knowledge bases learned by FRULER include TSK fuzzy rules that accurately predict the temperature dynamics from a set of different predictors that can be measured both inside and outside the building.

II. FRULER: Fuzzy RUle Learning through Evolution for Regression

FRULER (Fuzzy RUle Learning through Evolution for Regression) [9] is a novel GFS that obtains accurate and simple linguistic TSK-1 fuzzy rule base models for regression problems. FRULER (Fig. 1) is composed of a new instance selection method for regression, a novel multi-granularity fuzzy discretization of the input variables, and an evolutionary algorithm that uses a fast and scalable method with Elastic Net regularization to generate accurate and simple TSK-1 fuzzy rules.

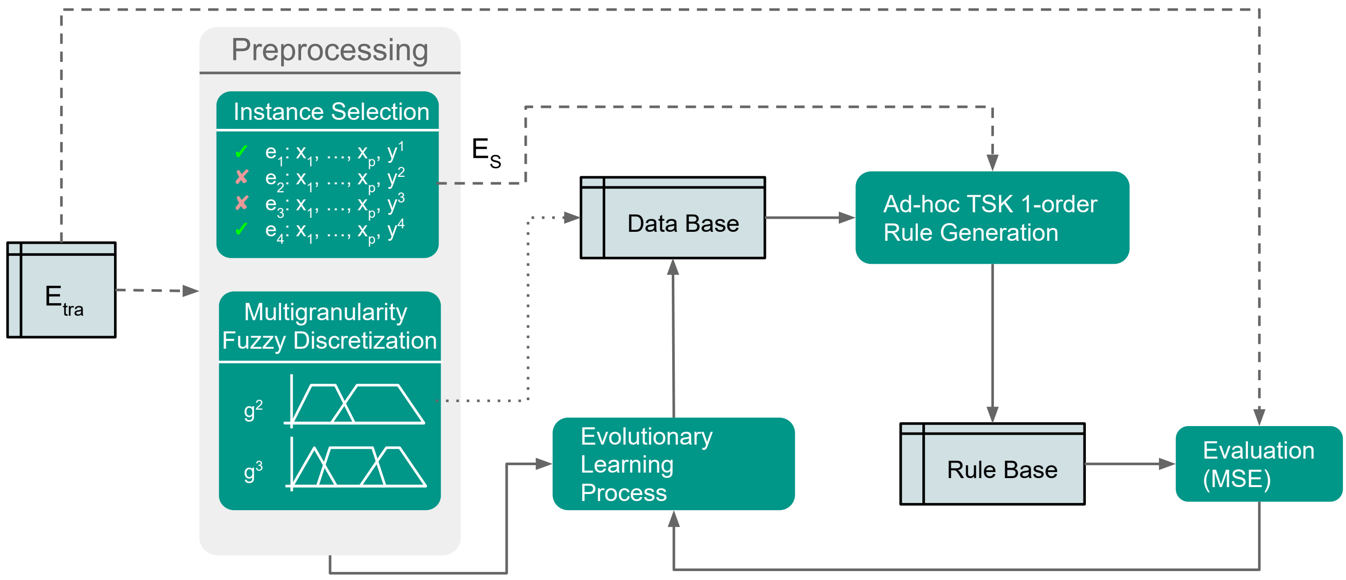

Figure 1: FRULER architecture. Dashed lines indicate flow of datasets, dotted lines multigranularity information and solid lines represent process flow.

A. *Instance selection*

The objective of the instance selection module is to reduce the variance of the models, focusing the generated rules on the representative examples. The instance selection method for regression is an improvement of the CCISR (Class Conditional Instance Selection for Regression) algorithm [8], which is an adaptation for regression of the instance selection method for classification CCIS (Class Conditional Instance Selection) [5].

The instance selection process is based on a relation called class conditional nearest neighbor (*ccnn*) [5], defined on pairs of points from a labeled training set as follows: for a given class c, $ccnn$ associates to instance a its $nearest\ neighbor$ computed among only those instances (excluded a) in class c. This relation, therefore, describes proximity information conditioned to a class label.

In regression problems, the outputs are real values instead of labels and, therefore, they must be discretized in order to use the *ccnn* relation. FRULER uses Kernel Density Estimation (KDE) with a gaussian kernel in order to estimate the probability density function of the output variable (y) in a non-parametric way. Once the probability density function is obtained, the local minimum determines the split points, and, therefore, which labels/classes are used for the *ccnn* relation. Thus, each instance is associated with one of the labels obtained by this process, and the instance selection method can follow the CCIS procedure.

B. *Multi-granularity fuzzy discretization*

In a multi-granularity proposal, each granularity has a different fuzzy partition. The generation of the fuzzy linguistic labels can be divided into two stages. First, the variable must be discretized to obtain a set of split points C^g for each granularity g. Then, given the split points, the fuzzy labels can be defined for each granularity. In a top-down approach, the split points are searched iteratively, i.e., only a new split point is added at each step, obtaining two new intervals. Therefore, the approach followed by FRULER aims to preserve interpretability between contiguous granularities: adding a new label to the previous granularity and modifying the flanks of the adjacent labels. In regression problems (TSK-1 in our case), the discretization process must search for the split point that minimizes the error when a linear model is applied to each of the resulting intervals.

In order to select the maximum number of split points for a variable, we used the well-known Bayesian Information Criterion (BIC). FRULER measures the error for BIC with a least squares model fitted for each interval of the discretization, and evaluates the complexity of the model as the number of inner splits and parameters fitted by the regression of each interval. After the discretization of a variable for each granularity, FRULER applies the method proposed in [3] in order to get the multi-granularity fuzzy partitions.

C. *Evolutionary algorithm*

The evolutionary algorithm learns a linguistic TSK model. The integration of the evolutionary algorithm with the preprocessing stage is as follows (Fig. 1):

- First, the instance selection process is executed over the training examples E_{tra} in order to obtain a subset of representative examples E_S.
- Then, the multi-granularity fuzzy discretization process obtains the fuzzy partitions for each input variable.
- Finally, the evolutionary algorithm searches for the best data base configuration using the obtained fuzzy partitions, generates the entire linguistic TSK rule base using E_S and evaluates the different rule bases using E_{tra}.

1) Chromosome Codification: The chromosome is codified with a double coding scheme ($C = C_1 + C_2$). C_1 represents the granularity of each input variable. C_2 represents the lateral displacements of the split points of the input variables fuzzy partitions.

2) Initialization: The initial pool of individuals is generated by a combination of two initialization procedures. A half of the individuals are generated with the same random granularity for each variable, while the other half is created with a different random granularity for each variable. The lateral displacements are initialized to 0 in all cases.

3) TSK Rule Base Generation: FRULER uses the Wang & Mendel algorithm to create the antecedent part of the rule base for each individual. The consequent part of the rules is learned using the Elastic Net method [14] in order to obtain the coefficients of the degree 1 polynomial for each rule. Elastic Net linearly combines the ℓ_1 (Lasso regularization) and ℓ_2 (Ridge regularization) penalties of the Lasso and Ridge methods, minimizing the following equation:

$$\hat{\beta} = \arg\min_{\beta} ||Y - X \cdot \beta||_2^2 + \lambda \cdot \alpha \cdot \|\beta\|_2^2 + \lambda \cdot (1-\alpha) \cdot \|\beta\|_1 \quad (1)$$

where β is the coefficients vector, Y is the outputs vector, X is the inputs matrix, λ is the regularization parameter and α represents the trade-off between ℓ_1 and ℓ_2 penalization. In order to solve the minimization problem of Elastic Net (Eq. 1), we used Stochastic Gradient Descent (SGD).

The rule base is generated using only those examples in E_s. In this manner, those examples that are not representative are not taken into account, the method avoids the generation of too specific rules, and reduces the time needed to create the rule base.

4) Evaluation: The fitness function is:

$$\mathit{fitness} = MSE(E_{tra}) = \frac{1}{2 \cdot |E|} \sum_{i=1}^{|E|} (F(x^i) - y^i)^2, \quad (2)$$

where E_{tra} is the full training dataset and $F(x^i)$ is the output obtained by the knowledge base for input x^i. Using all the examples for evaluation can be seen, in some way, as a validation process, as the rule base was constructed with a subset of them (E_S).

5) Selection and Replacement: The selection is performed by a binary tournament. On the other hand, the replacement method joins the previous and current populations, and selects the N best individuals as the new population.

6) Crossover and Mutation: FRULER has two crossover operations: one-point crossover for exchanging the C_1 parts (it also exchanges the corresponding C_2 genes) and, when the C_1 parts are equal, the parent-centric BLX (PCBLX) is used to crossover the C_2 part.

The mutation (with probability p_{mut}) applies two possible operations with equal probability to a randomly selected gene of the C_1 part: i) decreasing the granularity by 1 or ii) increasing the granularity to a more specific granularity —all the granularities have the same chance. In order to calculate the new lateral displacements in the corresponding C_2 part, the displacements of the previous granularity are taken into account.

7) Local Search: After the replacement, all the new individuals go to a local search process. This stage generates n_{ls} new C_1 parts with equal or less granularity for each variable, and the C_2 part is generated randomly. If there is a solution that obtains better fitness, then it replaces the original individual.

8) Restart and Stopping Criteria: The restart mechanism uses the incest prevention threshold L as a trigger. At each iteration, L is decreased in a value that depends on if there are no new individuals or the best individual does not change. When L reaches 0, the population is restarted. When the restart criterion is fulfilled twice or the number of evaluations reaches a threshold, the algorithm stops.

III. Indoor temperature prediction

The main goal of the OPERE project [4] is to implement efficient management systems in both thermal and electrical energy grids in existing installations with large energy consumption. To achieve this goal, in this work we propose a method that automatically learns an accurate and interpretable non-linear model using FRULER. The learned models predict the indoor temperature dynamics of an existing building in order to find a better heating control that minimizes the energy consumption without sacrificing thermal comfort. Concretely, we focus this study on the residential facilities of Monte da Condesa, a building located at the University of Santiago de Compostela.

Monte da Condesa comprises a set of centers that act as separate buildings, but nevertheless maintain thermal interaction through their conditioning circuits connected to a common cogeneration plant. The building is about 25,000 m^2 and reached in 2013 a total power consumption of 5,747 MWh. The set of all centers is supervised by a SCADA system that has more than 469 variables (inputs and outputs) that are associated with signals from the primary heating circuits and power consumption. Signals are collected in two different ways: synchronous (sync) and asynchronous (async). Synchronous signals are sequentially sampled at a fixed interval of 10 s, whereas asynchronous signals are registered by detecting a change of a value above an stablished threshold. These signals include information about the indoor temperature of each floor, the outside temperature, the water temperature of the pump water heating systems, plus many other low level variables. In order to predict the indoor temperatures, we focus on the variables that may directly affect the temperature dynamics. These variables are represented in Fig. 2, which shows a high-level representation of the building. T_{in}^n corresponds with the indoor temperature sensors of the building. In total, there are 6 different sensors ($T_{in}^0, \ldots, T_{in}^5$), one for each floor, which are the objective variables we want to predict. T_{flow1} and T_{flow2} refer to the temperature of the pumped water of the two heating systems installed in Monte da Condesa. T_{flow1} corresponds with the pump water heating system that feeds both floors 0 and 1, whereas T_{flow2} feeds the remaining floors.

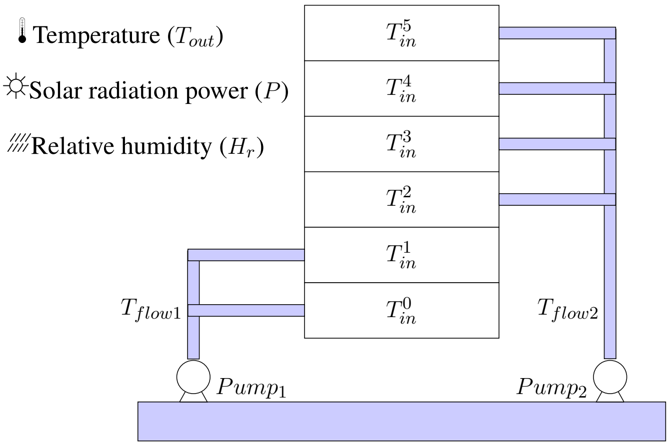

Figure 2: Schema of the Monte da Condesa Residence with the related variables.

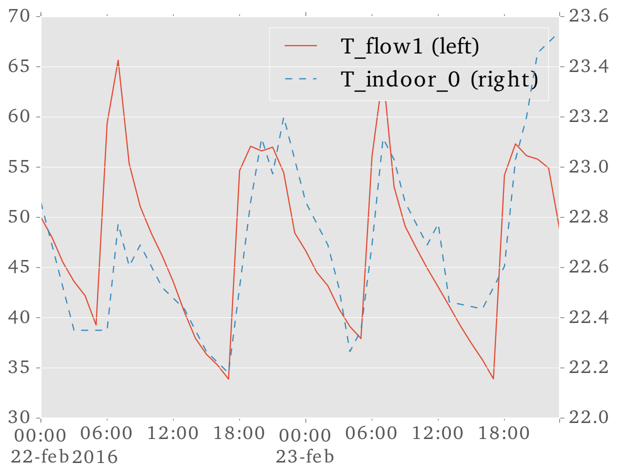

Figure 3: T_{flow1} temperature (left) vs T_{in}^0 indoor temperature (right).

Fig. 3 shows an example of T_{flow1} and T_{in}^0 between 22-02-2016 and 24-02-2016.

In addition to these SCADA variables, we also obtained the humidity H_r and solar radiation power P from *Santiago-EOAS*, a Meteogalicia [7] weather station situated approximately 100 meters from the reference building. Synchronous measures were downsampled to 1 h bins and asynchronous measures were converted into time series by appling linear interpolation and 1 h resampling. To summarize, the selected signals, sampled at 1 h interval (t) are:

- $T_{in}^n(t)$: indoor temperature at t of floor n (°C, async).
- $T_{out}(t)$: outside temperature at t (°C, async).
- $T_{flow1}(t)$: water temperature of the first heating system (1) at t (°C, sync).
- $T_{flow2}(t)$: water temperature of the second heating system (2) at t (°C, sync).
- $H_r(t)$: relative humidity (%, sync, Meteogalicia).
- $P(t)$: global solar radiation power (W/m^2, sync, Meteogalicia).
- T_{out}^d: calculated as $T_{out}^d(t+k) = T_{out}(t+k) - T_{out}(t)$ includes information about the trend of the outside temperature in the given interval.

Note that, for the sake of clarity, in the following we will refer to T_{flow} instead of T_{flow1} and T_{flow2}, where $T_{flow} = T_{flow1} \forall n \in [0,1]$ and $T_{flow} = T_{flow2} \forall n \in [2,5]$. T_{flow} is used to calculate other variables, such as the boiler operating percentage ($\%r$) and the time since the boiler stopped working (t_{stop}):

- $\%r$: it is calculated, from T_{flow} as the percentage the boiler is working in a time interval. It is assumed that the boiler is not working when the temperature is decreasing.
- t_{stop}: this variable represents the time from the moment the boiler stopped working to the prediction time. If the boiler is still working, $t_{stop} = 0$.

All these features were used to construct different rule-based regression models F with FRULER to predict each variable response $\hat{T}_{in}^n(t+k), n \in [0,5]$ for different values of k (hours ahead in time), where $\hat{T}_{in}^n$ is the predicted indoor temperature on floor n at instant $t+k$. $T_{in}^n(t)$, $T_{out}(t)$, T_{out}^d, $H_r(t+k)$, and $P(t+k)$ are used in all models. Note that the values of relative humidity and global radiation are set for the prediction time. We propose three different models (Fig. 4):

- *Model 1*: T_{flow} is hourly-averaged, so that four predictors ($k=4$) are considered ($T_{f_{t0}}, \ldots, T_{f_{t3}}$):

$$\hat{T}_{in}^n(t+k)_{M1} = F[T_{flow}(t), T_{in}^i(t), T_{out}(t), P(t+k), H_r(t+k), T_{out}^d(t+k), T_{f_{t0}}, \ldots, T_{f_{t3}}]$$

- *Model 2*: every 10 minutes, a value of the boiler operation is calculated from the related flow temperature. Thus, ($\%r_0, \ldots, \%r_3$) are the hourly-averaged result of this binary response:

$$\hat{T}_{in}^n(t+k)_{M2} = F[T_{flow}(t), T_{in}^n(t), T_{out}(t), P(t+k), H_r(t+k), T_{out}^d(t+k), \%r_0, \ldots, \%r_3]$$

- *Model 3*: this model intends to reduce the number of input variables. Therefore, the boiler operating percentage ($\%r$) represents the working time for the full interval. Despite the loss of information, t_{stop} is introduced to give a better comprehension of the boiler behaviour:

$$\hat{T}_{in}^n(t+k)_{M3} = F[T_{flow}(t), T_{in}^n(t), T_{out}(t), P(t+k), H_r(t+k), T_{out}^d(t+k), \%r, t_{stop}]$$

IV. Experiments and Results

A. Experimental setup

FRULER was designed to keep the number of parameters as low as possible. For the instance selection technique, no parameters are needed. In the multi-granularity fuzzy discretization, the fuzziness parameter used for the generation of the fuzzy intervals from the split points was 1, i.e., the highest fuzziness value. For the evolutionary algorithm, the values of the parameters were: population size = 61, maximum

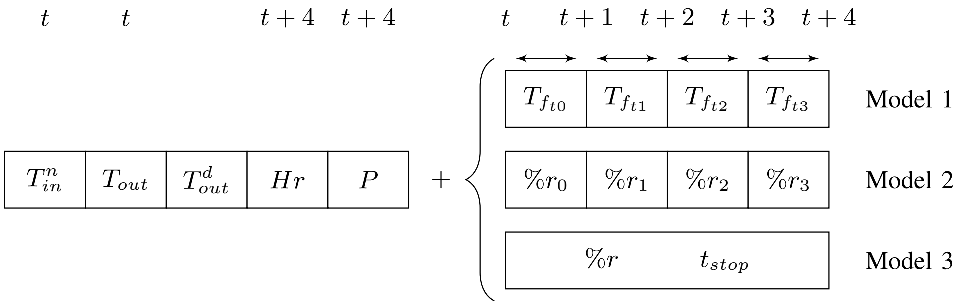

Figure 4: Graphical representation of the tested models.

	Model 1		Model 2		Model 3	
Floor	#Rules	Test Error	#Rules	Test Error	#Rules	Test Error
P0	9,1	**0,46**	**5,1**	0,50	11,1	0,54
P1	8,5	0,35	**6,0**	**0,32**	7,6	**0,32**
P2	6,2	**0,47**	**4,6**	0,48	8,4	**0,47**
P3	5,6	0,56	5,2	**0,54**	**4,8**	**0,54**
P4	5,8	0,43	**4,8**	0,41	6,5	**0,40**
P5	7,5	0,30	7,0	**0,28**	**5,9**	**0,28**

(a) Average number of rules (#Rules) and test error in °C (Test Error) for the compared models.

Model	Ranking
Model 3	1.667
Model 2	1.917
Model 1	2.417
p-value	0.455

(b) Friedman Test.

Table I: Experimental results of the three models.

number of evaluations = 100,000, $p_{cross} = 1.0$, $p_{mut} = 0.2$, and $n_{ls} = 5$. For the generation of the TSK fuzzy rule bases, the weight of the tradeoff between ℓ_1 and ℓ_2 regularizations on the Elastic Net is $\alpha = 0.95$, and the regularization parameter λ was obtained from a grid search in the interval $[1, 1E-10]$. η^0 was obtained halving the initial value (0.1) until the result worsens.

A 5-fold cross validation was used in all the experiments. Moreover, 6 trials (with different seeds for the random number generation) of FRULER were executed for each 5-fold cross validation. Thus, a total of 30 runs were obtained for each model. The results shown in the next section are the mean values over all the runs. Data was recorded from 25-01-2016 to 20-05-2016 (2,754 h).

B. Results

Table Ia shows the average results of FRULER for the three learned models. For each model and floor, the table displays the number of rules of the learned knowledge base, and the test error measured in °C. These indicators allow to compare both the simplicity and the accuracy of the learned models. The values with the best accuracy —lowest error— and best number of rules —lowest value— in Table I are marked in bold.

Model 2 gets the lowest number of rules in four of the six floors. Nonetheless, the accuracy is similar for all models. In order to check whether there are significant differences among the models, we applied the Friedman statistical test, that computes the ranking of the results of the algorithms. The application of the test, using the STAC platform [10], rejects the null hypothesis which states that the results of all the algorithms are equivalent with a given confidence significance level ($\alpha = 0.05$). Table Ib shows the ranking for the test error and the p-value of the test, which indicates that the differences among the models are not statistically significant.

V. Conclusions

In this paper we presented three different models for indoor temperature prediction using the FRULER Genetic Fuzzy System to generate the knowledge base, made up of TSK fuzzy rules. The models have been learned from data recorded at Monte da Condesa Residential College during 2,754 hours and from several sensors. The models can predict the future indoor temperature for each floor of the building with an average error in the range 0.28-0.50 °C. The learned models will be used in the near future in the LIFE-Opere EU project [4] for planning efficient heating control strategies, predicting the indoor temperature in order to guarantee that the global power consumption of the heating system is reduced without sacrificing thermal comfort.

References

[1] Prediction of building's temperature using neural networks models. *Energy and Buildings*, 38(6):682–694, 2006.

[2] Aurélie Foucquier, Sylvain Robert, Frédéric Suard, Louis Stéphan, and Arnaud Jay. State of the art in building modelling and energy performances prediction: A review. *Renewable and Sustainable Energy Reviews*, 23:272–288, 2013.
[3] Hisao Ishibuchi and Takashi Yamamoto. Performance evaluation of fuzzy partitions with different fuzzification grades. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE).*, volume 2, pages 1198–1203. IEEE, 2002.
[4] Life-OPERE. Life-OPERE project web page, http://www.life-opere.org/, Last visited May 30th 2016.
[5] Elena Marchiori. Class conditional nearest neighbor for large margin instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):364–370, 2010.
[6] A. Mechaqrane and M. Zouak. A comparison of linear and neural network ARX models applied to a prediction of the indoor temperature of a building. *Neural Computing and Applications*, 13(1):32–37, 2004.
[7] Meteogalicia. Galician meteorological web page, http://www.meteogalicia.es/, Last visited May 30th 2016.
[8] I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín. An instance selection algorithm for regression and its application in variance reduction. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2013.
[9] I. Rodríguez-Fdez, M. Mucientes, and A. Bugarín. FRULER: Fuzzy rule learning through evolution for regression. *Information Sciences*, 354:1–18, 2016.
[10] Ismael Rodríguez-Fdez, Adrián Canosa, Manuel Mucientes, and Alberto Bugarín. STAC: a web platform for the comparison of algorithms using statistical tests. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2015.
[11] Tao Lu and M Viljanen. Prediction of indoor temperature and relative humidity using neural network models: model comparison. *Neural Computing Applications*, 18(4):345–57, 2009.
[12] Catalin Teodosiu, Raluca Hohota, Gilles Rusaouën, and Monika Woloszyn. Numerical prediction of indoor air humidity and its effect on indoor environment. *Building and Environment*, 38(5):655–664, 2003.
[13] Bertil Thomas and Mohsen Soleimani-Mohseni. Artificial neural network models for indoor temperature prediction: investigations in two buildings. *Neural Computing and Applications*, 16(1):81–89, 2006.
[14] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, 67(2):301–320, 2005.